Multicast (Group) Addresses for Layer 2 (Ethernet) Transport of IEEE 1588 PTP Messages, with Application to AVB

Geoffrey M. Garner
SAMSUNG Electronics (Consultant)

IEEE 802.1 AVB TG
IEEE 1588
2006.05.16

gmgarner@comcast.net

Outline

- Introduction
- Goals of presentation
- Terminology
- IEEE 1588 node types and desired forwarding behavior for each PTP message
  - Subset of node types and messages used in AVB
- IEEE 802.1 reserved multicast addresses
- Generating and forwarding rules for PTP messages, for each PTP node type

Introduction

- At present, all addresses in IEEE 1588 Version 2 will be multicast by default
  - The only exception to this will be for communication within fault-tolerant master clusters, which will use unicast addresses
  - Note that IEEE 1588 Version 1 uses only multicast addressing
- Unicast addresses will be available for at least some messages as an option
- The AVB synchronization network architecture is (see [1] for details)
  - AVB networks will consist of a single Grandmaster (GM) that synchronizes slave clocks through Peer-to-Peer (P2P) Transparent Clocks (TCs)
    - The AVB cloud will be a single IEEE 1588 subdomain
  - The GM and Slave Clocks will be Ordinary Clocks (OCs)
  - Each OC (both GM and slaves) will be collocated with a P2P TC
    - I.e., an AVB node will consist of a P2P TC and collocated OC
  - AVB networks will not use IEEE 1588 Boundary Clocks (BCs)

Introduction (Cont.)

- IEEE 1588 Precision Time Protocol (PTP) messages in AVB networks will be multicast
  - With multicast addressing for Sync and Follow_Up, the GM only needs to send one Sync and one Follow_Up message at each sync interval
  - If the addressing were unicast, the GM would have to send a separate Sync and Follow_Up to each slave clock, i.e., to each AVB network node
    - The GM would not scale well in this case with increasing network size
  - In addition, each AVB network node will exchange Pdelay messages with the neighboring node on each link to measure propagation time on that link
    - Multicast addressing will simplify this process, as the same address can be used in every Pdelay message at every node
      - To make this work, we need to ensure that Pdelay messages are not forwarded (this is described more fully in the following slides; here we are only motivating why multicast addressing is desirable for AVB networks)
- So far, there has been little discussion in IEEE 1588 on the details of how Layer 2 Ethernet multicast addressing will work
  - There has been some discussion of this in recent AVB calls

Goals of Presentation

- Summarize the current understanding (of the author) of how multicast addressing for IEEE 1588 PTP messages transported over Layer 2 Ethernet can work, and make any necessary corrections during discussion
  - Understanding is based on the limited discussion in recent AVB calls
- Attempt to determine which aspects will be specified in IEEE 1588 and which aspects in IEEE 802.1as
  - If it is determined that material not needed for AVB is also beyond the scope of IEEE 1588 (e.g., details of the actions E2E TCs take when receiving PTP messages that use multicast addresses), it must be determined where this material will go
    - E.g., also in 802.1as; in another document?
    - It seems desirable to have a single method for multicast addressing for all applications that use IEEE 1588 over Layer 2 Ethernet
- Note that the material here does not preclude non-AVB applications from using unicast addressing; this presentation is simply limited to describing multicast addressing

Terminology

- IEEE 1588 refers to the “forwarding” of various PTP messages (e.g., Sync, Follow_Up, etc.)
  - Forwarding occurs after possibly altering the PTP payload
    - By “PTP payload,” we mean the Ethernet payload, i.e., everything between (and not including) the Ethertype and FCS fields
- At the Ethernet (IEEE 802.1) layer, these messages are not forwarded by participating (i.e., by 1588 enabled or AVB enabled) bridges
  - Rather, the Ethernet bridge blocks the message and passes it to the PTP layer
  - The PTP layer Processes the message and, based on the processing, does one of several things
    - Sends one or more related messages of the same type on one or more ports
      - Related messages may essentially be the original message with one or more fields altered
        - E.g., may send Sync messages on several ports, with the PTP payload of each Sync message sent identical to the PTP payload of the received Sync message except for the addition of the residence time (for each respective message sent) to the correction field
    - Responds to the received message with a different PTP message
    - Does not send any messages
- However, these messages are forwarded at the Ethernet layer by non-participating (non-1588) bridges
- In this presentation, we try to be clear in using the term “forwarding” whether it is at the Ethernet layer or the PTP layer

Node Types and Desired Forwarding Behavior

- Types of PTP nodes
  - Ordinary clocks
  - Boundary clocks (not used in AVB)
  - Peer-to-Peer (P2P) transparent clocks
  - End-to-End (E2E) transparent clocks (not used in AVB)
- More precisely, the above actually are functions
  - It is possible to combine an OC and TC (either P2P or E2E) function in a single node
  - An AVB node will consist of an OC and P2P TC function (see [1])
- PTP messages
  - Sync
  - Follow_Up
  - Delay_Req (not used in AVB)
  - Delay_Resp (not used in AVB)
  - Announce
  - Pdelay_Req
  - Pdelay_Resp
  - Pdelay_Resp_Follow_Up

Node Types and Desired Forwarding Behavior

- BCs and OCs do not forward any of the above PTP messages
  - A BC or OC that is in the master state sends Sync and, if follow-up capable, Follow_Up
  - A BC or OC that is in the slave state processes received Sync and Follow_Up messages
  - A BC or OC that is in the slave state send Delay_Req to the master BC or OC on each communication path
  - A BC or OC that is in the master state responds to Delay_Req received from each slave TC on the communication path with Delay_Resp
  - BCs and OCs send Announce messages and process received Announce messages, based on executing the Best Master Clock (BMC) algorithm
  - BCs and OCs send Pdelay_Req messages on links that lead to P2P TCs (possibly with one or more E2E TCs in between)
  - BCs and OCs respond to Pdelay_Req messages with Pdelay_Resp and possibly Pdelay_Resp_Follow_Up messages, on links that lead to P2P TCs (possibly with one or more E2E TCs in between)

Node Types and Desired Forwarding Behavior

- E2E TCs forward all of the above PTP messages
  - Depending on the message type and whether the E2E TC is on-the-fly or follow-up, the E2E TC may or may not alter the message
    - The alteration would generally be the addition of a residence time to a correction field, if needed (Announce messages are not altered)
    - See [2] for details
- P2P TCs forward Sync, Follow_Up, and Announce messages
  - Depending on the message type and whether the P2P TC is on-the-fly or follow-up, the P2P TC may or may not alter the message
    - The alteration would generally be the addition of a residence time to a correction field, if needed (Announce messages are not altered)
    - See [2] for general details, and [1] for details specific to AVB networks
- P2P TCs do not forward Pdelay_Req, Pdelay_Resp, and Pdelay_Resp_Follow_Up messages

Node Types and Desired Forwarding Behavior

- P2P TCs will not process Delay_Req and Delay_Resp messages
  - Therefore, BCs and OCs attached to P2P TCs must process Pdelay messages
    - AVB network nodes will process Pdelay messages, and AVB networks will not use Delay_Req and Delay_Resp
  - For general IEEE 1588 networks, this must be confirmed in the TC Subcommittee
    - there had been previous discussion that BCs and OCs would process Delay_Req and Delay_Resp for backward compatibility with IEEE 1588 Version 1
    - This issue is not relevant to AVB

IEEE 802.1 Reserved Multicast Addresses

- The OUI 00-80-C2 is assigned to IEEE 802.1
  - This includes unicast addresses (that begin with 00-80-C2) and multicast addresses (that begin with 01-80-C2)
- IEEE 802.1D-2004 defines in Table 7-10 Reserved Multicast Addresses 01-80-C2-00-00-00 through 01-80-C2-00-00-0F
  - The first 4 (i.e., those that end in –01, –02, –03, and –04) are assigned
  - The rest are reserved for future standardization
  - Frames sent to any of these addresses are never forwarded by bridges
- IEEE 802.1D-2004 defines in Table 12-1 GARP Application Addresses 01-80-C2-00-00-20 through 01-80-C2-00-00-2F
  - The 2 (i.e., those that end in –01 and –02) are assigned
  - The rest are reserved for future standardization
  - Bridges that support GARP applications do not forward frames with these addresses; instead, they process them
  - Bridges that do not support GARP applications do forward frames with these addresses

IEEE 802.1 Reserved Multicast Addresses

- If a Reserved Multicast Address were defined for PTP messages, then Ethernet bridges would never forward these messages at the Ethernet layer
  - Instead, any frame sent to that reserved multicast address would be passed to the PTP layer for processing
- A single Reserved Multicast Address from the range in 802.1D, Table 7-10, can be assigned for PTP messages
  - Use one of the currently unassigned addresses
- Alternatively, if it is more convenient, PTP messages can use a new Reserved Multicast Address defined by IEEE 802.1 (taken from the 00-80-C2 OUI, i.e., beginning with 01-80-C2)
- Any frames sent to this address will be blocked by a bridge and passed to the PTP layer for processing
- This scheme will work as long as all bridges are participating (i.e., 1588 enabled)
  - With this scheme, a non-1588 bridge would also block the PTP messages; they would then be lost because the bridge would have no PTP layer to pass them to
- Only need one reserved address, because Ethernet layer simply recognizes the address and passes the message to the PTP layer; PTP layer does all the processing

IEEE 802.1 Reserved Multicast Addresses

- If non-participating (i.e., non-1588) bridges are present, then the PTP messages should use a reserved address with forwarding behavior analogous to those of the GARP Application Addresses in 802.1D, Table 12-1
- A single unused address can be taken from one of the addresses owned by IEEE 802.1 (taken from the 00-80-C2 OUI, i.e., beginning with 01-80-C2) and assigned to be used as a "PTP Application Address"
- If a bridge is participating (i.e., 1588 enabled, which includes AVB-enabled), it should not forward (at the Ethernet layer) frames sent to this address
  - Instead, the frames will be passed to the PTP layer
- If a bridge is not participating (i.e., not 1588 enabled, which means not AVB enabled), it should forward frames sent to this address on all non-blocked ports excluding the port the frame arrived on
  - If there are no non-blocked ports other than the port the frame arrived on, the frame is dropped
  - This ensures that the PTP frames reach 1588 enabled bridges when non-1588 enabled bridges are present
- Only need one reserved address, because Ethernet layer simply recognizes the address and either forwards the frame or passes the message to the PTP layer; PTP layer does all the processing

Forwarding Rules for PTP Messages

- BCs and OCs will operate as indicated on Slide 8; they do not forward (at the PTP layer) any of the above PTP messages
  - A BC or OC sends (i.e., originates) Sync and, if follow-up capable, Follow_Up on each port that is in the master state
    - This means that Sync and Follow_Up are not sent on ports that are:
      - Blocked at the Ethernet layer by STP (or any equivalent protocol used at the Ethernet layer to ensure there are no loops)
      - Not blocked at the Ethernet layer but are in the slave state (note that an OC has only one port, and therefore the OC is either a master or a slave)
      - Not blocked at the Ethernet layer but in the PTP-passive state; this means the port is blocked at the PTP layer
    - The PTP Best Master Clock (BMC) algorithm is able to produce a spanning tree for the BC and OC (but not TC) nodes at the PTP layer
      - For layer 2 Ethernet operation, a spanning tree may already be produced by STP or an equivalent protocol at layer 2
      - This issue is not relevant for AVB, because AVB will not use BCs
  - A BC or OC processes Sync and Follow_Up messages received on the single port that is in the slave state (and uses the information to synchronize to its master)
    - Sync and Follow_Up messages will not be received on ports blocked at the Ethernet layer, nor ports that are in the PTP-passive state, i.e., blocked at the PTP layer

Forwarding Rules for PTP Messages

❑BCs and OCs (Cont.)

- A BC or OC sends Delay_Req to the master BC or OC on the single port that is in the slave state
  - Delay_Req is not sent on ports blocked at the Ethernet layer, nor ports in the PTP-passive state or master state
- A BC or OC responds to Delay_Req received from each slave TC on each port that is in the master state with Delay_Resp
  - Delay_Resp is not sent on ports blocked at the Ethernet layer, nor ports in the PTP-passive state nor slave state
- BCs and OCs send Announce messages and process received Announce messages, based on executing the Best Master Clock (BMC) algorithm
  - Announce messages are not sent and not received on ports blocked at the Ethernet layer

Forwarding Rules for PTP Messages

❑BCs and OCs (Cont.)

- BCs and OCs send Pdelay_Req messages on links that lead to P2P TCs (possibly with one or more E2E TCs in between)
    - Pdelay_Req messages are sent on all ports that lead to P2P TCs, regardless of whether they are in the master, slave, or PTP-passive states, or whether they are blocked at the Ethernet layer
    - This is because propagation delay must be measured on all links and made available in the event of reconfiguration
- BCs and OCs respond to Pdelay_Req messages with Pdelay_Resp and possibly Pdelay_Resp_Follow_Up messages, on links that lead to P2P TCs (possibly with one or more E2E TCs in between)
    - As above, Pdelay_Resp messages are received on all ports that lead to P2P TCs, regardless of whether they are in the master, slave, or PTP-passive states, or whether they are blocked at the Ethernet layer
    - This is because propagation delay must be measured on all links and made available in the event of reconfiguration

Forwarding Rules for PTP Messages

- P2P and E2E TCs will process received Sync, Follow_Up, and Announce messages in accordance with the PTP protocol
- They will then send new (i.e., possibly modified based on the received message) Sync, Follow_Up, or Announce messages on all ports that are not blocked by STP (or any equivalent protocol used at the Ethernet layer to ensure there are no loops), except the port the message arrived on
- E2E TCs will also process Delay_Req and Delay_Resp and then send new Delay_Req and Delay_Resp messages on non-blocked ports except the port the message arrived on
- P2P TCs will not process Delay_Req and Delay_Resp (this must be confirmed by the IEEE 1588 TC Subcommittee, but in any case is not relevant for AVB; see slide 10)

Forwarding Rules for PTP Messages

- P2P TCs will generate and send Pdelay_Req on all ports, including blocked ports
  - Pdelay messages are sent on blocked ports so that propagation delays can be measured on these links and made available in the event of reconfiguration
- P2P TCs will respond to Pdelay_Req only on the port that the Pdelay_Req arrived on; the response will be with Pdelay_Resp and possibly Pdelay_Resp_Follow_Up
- E2E TCs will process received Pdelay_Req, Pdelay_Resp, and Pdelay_Resp_Follow_Up messages and then send new (i.e., modified) messages on all ports that are not blocked except the port the message arrived on
  - If there are no such ports, then nothing is sent by the E2E TC
  - This handling by E2E TCs ensures that a Pdelay message sent from one P2P TC to another through a network of E2E TCs will reach its destination; note that some spurious Pdelay messages may reach E2E TCs that aren't in the path to the destination P2P TC, but they will simply be dropped
  - This handling does not solve the 1:N problem (i.e., case where more than 2 P2P TCs are connected through one or more E2E TCs, but leaves open the possibility of solving it in the future.

References

1. Geoffrey M. Garner, *Description of Use of IEEE 1588 Peer-to-Peer Transparent Clocks in A/V Bridging Networks*, Revision 1.1, April 24, 2006.
2. *Transparent Clock – Working Technical Description, Revision 13*, IEEE 1588 Transparent Clock Subcommittee, February 17, 2006.